

# Auxiliary Problem Methods with Varying Core Functions for Large-Scale Structured Convex Problems

Lei Zhao\*, Daoli Zhu†, Bo Jiang‡

April 6, 2016

## Abstract

The auxiliary problem principle (APP), proposed by Cohen (1978, 1980), Cohen and Zhu (1984), aims to find the solution of an optimization problem through a sequence of auxiliary problems. The merits of this approach are two folds. First, the core function is usually separable, which makes the subproblems at each step decomposable and particularly attractive for parallel computing. Second, the choice of the core function is quite flexible. Consequently, by carefully specifying this function, APP may reduce to many standard optimization algorithms. In this paper, we pursue enhancing such flexibility by allowing the core function to be non-identical at each step of the algorithm, and name it varying auxiliary problem principle (VAPP). The convergence of VAPP for convex problem with coupling constraints is proved. Moreover, if this function is specialized to be quadratic, an  $o(1/k)$  convergence rate can be established. Interestingly, the new VAPP framework can cover several variants of Jacobian type ADMM as special cases. In the absence of the proximal terms, the condition used in our proof is weaker than that required by other literature for ADMM. Furthermore, our technique works for the convex problem with nonseparable objective and coupled linear constraints, which usually can not be handled by ADMM. Numerical results are presented to illustrate the efficiency of the proposed new framework.

**Keywords:** Auxiliary Problem Principle, Varying Core Function, Convergence Rate, Parallel and Distributed Computing.

---

\*Sino-US Global Logistics Institute and Antai College of Economics and Management, Shanghai Jiao Tong University, 200030 Shanghai, China([l.zhao@sjtu.edu.cn](mailto:l.zhao@sjtu.edu.cn))

†Sino-US Global Logistics Institute and Antai College of Economics and Management, Shanghai Jiao Tong University, 200030 Shanghai, China([dlzhu@sjtu.edu.cn](mailto:dlzhu@sjtu.edu.cn))

‡Research Center for Management Science and Data Analytics, School of Information Management and Engineering, Shanghai University of Finance and Economics, Shanghai 200433, China([isybojiang@gmail.com](mailto:isybojiang@gmail.com))

# 1 Introduction

The auxiliary problem principle (APP) for classical optimization problems was originated by Cohen [12, 13] and Cohen and Zhu [14]. This algorithm later turned out to be a generic framework that could cover quite a few optimization algorithms ranging from gradient (or subgradient) algorithms to decomposition/coordination algorithms as special cases by specifying various core functions. Moreover, the subproblem in each iteration of APP can be decomposed into a number of smaller problems that can be optimized separately. Thanks to this decomposable excellent numerical performance, APP has become the main theoretical basis of some parallel computing software such as DistOpt [15, 37]. Besides, APP also has wide applications in engineering systems, such as power systems [34, 39], multiple-robot systems [8]. In particular, this approach was adopted by Kim and Baldick to parallelize optimal power flow in very large interconnected power systems [34]; Ćela and Hamam applied this method to solve the optimal control problem of multiple-robot systems in the presence of obstacles [8].

In this paper, we consider the following block structured convex minimization problem with non-separable objective and coupled linear equality constraints:

$$\begin{aligned} \min \quad & (G + J)(u) := G(u_1, u_2, \dots, u_N) + \sum_{i=1}^N J_i(u_i) \\ \text{s.t} \quad & Au = \sum_{i=1}^N A_i u_i = b \\ & u_i \in U_i, \quad i = 1, 2, \dots, N. \end{aligned} \tag{1}$$

where each  $J_i$  is a convex but possibly nonsmooth function on  $U_i \subset \mathbf{R}^{n_i}$ , and  $A = (A_1, A_2, \dots, A_N) \in \mathbf{R}^{m \times n}$  is an appropriate partition of matrix  $A$  and  $A_i$  is an  $m \times n_i$  matrix,  $b \in \mathbf{R}^m$  is a vector. In fact, this problem was considered by Cohen and Zhu [14] and Zhu [45] as early as in 1983 in Hilbert space. Recently, due to its wide applications in signal processing, statistical learning, machine learning and bioinformatics (see [31] and references therein), problem (1) has regained some noticeable research attention [16, 22, 31]. Many solution methods of (1) are based on its associated augmented Lagrangian function:

$$L_\gamma(u, p) = (G + J)(u) + \langle p, Au - b \rangle + \frac{\gamma}{2} \|Au - b\|^2, \quad \text{with} \quad J(u) = \sum_{i=1}^N J_i(u_i). \tag{2}$$

One such example is the famous the alternating direction method of multipliers (abbreviated as ADMM), which usually however can not handle the coupling function  $G(\cdot)$  in the objective. The APP proposed by Cohen and Zhu [14] and Zhu [45] also belongs to this category. In particular, it can incorporate dual update in each iteration, and the auxiliary subproblem can be modified to account for the augmented Lagrangian function. In this regard, the specific updating scheme is

given by:

$$\begin{cases} u^{k+1} = \arg \min_{u \in U} K(u) - \langle \nabla K(u^k), u \rangle + \epsilon [\langle \nabla G(u^k), u \rangle + J(u) + \langle p^k + \gamma(Au^k - b), Au \rangle] \\ p^{k+1} = p^k + \rho(\sum_{i=1}^N A_i u_i^{k+1} - b), \end{cases} \quad (3)$$

where  $K(\cdot)$  is called the auxiliary core function with  $\epsilon$  being a positive number,  $\rho$  is the dual stepsize,  $U = U_1 \times \cdots \times U_N$  and  $\langle p^k + \gamma(Au^k - b), Au \rangle$  is obtained by linearizing the last two terms in the augmented Lagrangian function (2). If  $K(u) = \sum_{i=1}^N K_i(u_i)$ , then the subproblem in (3) becomes  $N$  independent subproblems on  $U^i$ . The auxiliary problem (3) provides a parallel decomposition scheme. Cohen and Zhu [14] and Zhu [45] showed that any cluster point of the sequence generated by this algorithm is the global optimizer as long as  $\rho$  and  $\epsilon$  are properly chosen. To our best knowledge, the convergence of the whole sequence and the convergence rate for the APP type algorithm have not been established yet in the previous works.

It is also interesting to note that optimal solution of the subproblem for updating  $u$  in (3) is unchanged when adding a constant term  $-K(u^k) + \langle \nabla K(u^k), u^k \rangle$ . Consequently, the objective involve a term  $K(u) - K(u^k) - \langle \nabla K(u^k), u - u^k \rangle$ , which is essentially the *Bregman distance* [6] of points  $u$  and  $u^k$ . In fact, this leads to the so-called general distance-like function, which has been widely used in the algorithm design. There are two popular choices for such general distance-like function including Bregman distance [11, 10, 43, 19, 33] and  $\varphi$ -divergence distance [43, 44]. In particular, Beck and Teboulle [2] showed that the mirror descent algorithm can be viewed as a nonlinear projected-subgradient type method with a general distance-like function. Recently, many works [3, 4, 42] explored the convergence rate of the algorithms when equipped with Euclidean distance. We note that, in all these works, the general distance-like function is invariant during the whole iterating procedure. However, the convergence property is unknown if the such distance-like function is allowed to be different in each iteration.

In this paper, we address this issue by proposing a new variant of APP with varying core function and name it *Varying Auxiliary Problem Principle (VAPP)*. Thus, at the  $k$ -th step, the core function is denoted as  $K^k$  to differentiate from the core functions used in other steps. Therefore, the updating scheme (3) is amended accordingly as

---

Varying Auxiliary Problem Principle (VAPP)

---

Initialize  $u^0 \in U$  and  $p^0$

**for**  $k = 0, 1, \dots$ , **do**

$$u^{k+1} = \arg \min_{u \in U} K^k(u) - \langle \nabla K^k(u^k), u \rangle + \epsilon [\langle \nabla G(u^k), u \rangle + J(u) + \langle p^k + \gamma(Au^k - b), Au \rangle] ;$$

$$p^{k+1} = p^k + \rho(\sum_{i=1}^N A_i u_i^{k+1} - b).$$

**end for**

---

The advantage gained by introducing such variability is the core function becomes much more flexi-

ble and the capability of establishing connections to many well known algorithms. One specification of interest is

$$K^k(u) = \frac{1}{2}(u - u^k)^\top D(u^k)(u - u^k), \text{ where } D(u^k) \text{ is an } n \times n \text{ matrix.}$$

Depending on the choice of  $D(u^k)$ , VAPP has connections to several standard algorithms [28, 46]. For instance, it is related to Newton's method when  $D(u^k) = \nabla^2 G(u^k)$  and it makes connection to Linearized Jacobi method by letting  $D(u^k) = M(u^k)$ , with  $M(u^k)$  being the diagonal part of  $\nabla^2 G(u^k)$ . Another possible choice of core function is

$$K^k(u) = \sum_{i=1}^N \nu_i G(u_1^k, \dots, u_{i-1}^k, u_i, u_{i+1}^k, \dots, u_n^k) + \sum_{i=1}^N \frac{\theta_i}{2} \|A_i u_i + (\sum_{j \neq i} A_j u_j^k - b)\|^2 + \sum_{i=1}^N \frac{\alpha_i}{2} \|P_i u_i\|^2, \quad (4)$$

which, we shall see later, leads to some variants of Jacobian ADMM when  $\nu_i = 0$ . Other alternatives of core functions will be discussed at the end of Section 3.

Interestingly, the relation of APP and ADMM was realized previously in [35], however no serious effort has been made to explore this relation carefully. In stead, a comparison [35] on the numerical performance of APP and ADMM was performed on several medium size power systems.

The standard ADMM can solve (1) with  $G(u) \equiv 0$ . In fact, this research can be traced back to 1976 by Gabay, Mercier, Glowinski and Marrocco [21, 23]. When there are only two block variables, the convergence of ADMM has been proved in [21, 23]. It is worth mentioning that the convergence can also follow from that of the so-called Douglas-Rachford operator splitting method [18, 25]. Although such convergence results have been established long time ago, the rate of convergence was only established recently by He and Yuan in [29], where they showed that the ADMM converges at the rate of  $O(1/k)$  with  $k$  being the number of total iterations. However, some negative results of this algorithm have also been discovered. In particular, Chen et al. [9] showed that ADMM fails to converge for a three-block linear feasibility problem. One way out of this dismay is to apply this algorithm to the scenario with more desirable properties. Indeed, by imposing the strong convexity condition to some parts of the objective, an  $O(1/k)$  convergence rate can still be guaranteed for multi-block ADMM [36, 7].

On the other hand, to accommodate parallel computation, a Jacobian type ADMM (abbreviated as JADM) and its proximal version (abbreviated as PJADM) have been considered by Deng et al. in [17]. Generally speaking, JADM is divergent even for a simple two-block linear feasibility problem provided by He et al. [30]. However, when matrices  $A_i$  are mutually near-orthogonal and have full column-rank, it can be shown that JADM is indeed convergent [17]. In addition, an  $o(1/k)$  convergence rate can be achieved for PJADM. As illustrated in Section 5, PJADM can be viewed as a special case of VAPP by specifying  $K^k(u)$  in the form of (4) with  $\nu_i = 0$ ,  $\epsilon = 1$ ,  $\theta_i = \gamma = \rho$ ,  $\alpha_i = 1$  and  $\bar{P}_i = P_i^\top P_i$  for all  $i$ , when  $G$  is not involved in the objective. There are certainly other

attempts to incorporate ADMM with parallel computation. For instance, He et al. [30] have studied some variants of Jacobian type ADMM by taking additional correction steps at every iteration.

Another related stream of work is the ADMM type algorithms which can handle the coupling term  $G(\cdot)$  in the objective; see [16, 22, 31]. The techniques that are used to cope with the non-separability can be summarized as two categories. The technique adopted in [16, 31], is to replace the nonseparable part of the augmented Lagrangian function by an easily solved upper-bound (it is also called majorization-minimization). The other one is to linearize the nonseparable part and then solve the revised subproblem. It has been shown in [22] that under certain conditions an  $O(1/k)$  convergence rate can still be assured. However, all of their algorithms can not be amendable for parallel computation.

## 1.1 Main contributions and summary of results

This paper delivers a few novel results from various perspectives. First of all, we extend the classical APP decomposition method of Cohen and Zhu [14] to accommodate the varying core functions in each iteration and refer to it as VAPP. This approach can serve as a framework that covers many distributed computing algorithms. Theoretically, we prove that as long as the core functions is strongly convex and Lipschitz continuous (possibly different in each iteration), the whole sequence generated by this approach will converge. Moreover, a convergence rate of  $o(1/k)$  for VAPP with quadratic varying core functions has been established. To our best knowledge, this is the first iteration complexity result for APP type algorithms.

Secondly, after equipping the core function in the form of (4), VAPP is closely related to the Jacobian type ADMM in [17]. For example, PJADM in [17] is one specializations of VAPP, and we can still achieve  $o(1/k)$  convergence rate without adding proximal term associated with block  $u_i$  if the corresponding matrix  $A_i$  has full column rank. Furthermore, if none of the proximal term is imposed, comparing to JADM of [17] the convergence of our algorithm does not require matrices  $A_i$  are mutually near-orthogonal.

The third contribution is that, our technique can handle convex optimization problem with *non-separable objective*. Although there are other works [16, 22, 31] are devoted to this topic, some additional conditions may be required. For example, without the proximal term, the strongly convexity of some  $J_i$ s is needed in [22]. Moreover, VAPP is suitable for parallel and distributed computing, which can not be incorporated into the algorithms proposed in [16, 22, 31]. Our numerical experiments show that the proposed VAPP algorithm has promising numerical efficiency.

The rest of this paper is organized as follows. Section 2 is devoted to the notations adopted and the assumptions that we make in this paper. In section 3 we carry out the convergence analysis of

VAPP with general varying core functions. After that, numerous instances of algorithms that fit our general framework are proposed. In section 4, an adaptively tuning strategy on parameter  $\epsilon$  is proposed to improve the practical performance of our algorithms. The  $o(1/k)$  convergence rate of VAPP with quadratic core function is presented in section 5. Numerical experiments are performed in section 6 and the results justify the efficiency of our methods. Finally, we end our paper with some conclusions.

## 2 Notations and Assumptions

In this paper, we let

$$u := \begin{pmatrix} u_1 \\ \vdots \\ u_N \end{pmatrix} \in \mathbf{R}^n, U = U_1 \times \cdots \times U_N \text{ and } A = [A_1, \cdots, A_N] \in \mathbf{R}^{m \times n}, \text{ where } n = \sum_{i=1}^N n_i.$$

We denote  $\langle \cdot \rangle$  and  $\| \cdot \|$  as the inner product and Euclidean norm of vector, respectively. For a matrix  $B \in \mathbf{R}^{\ell \times \ell}$ ,  $\|B\|$  stands for the spectral norm, which is the largest singular value of  $B$ . We use  $\lambda_{\max}(B)$  and  $\lambda_{\min}(B)$  to denote the maximum and minimum eigenvalue of  $B$  respectively.

For a positive semi-definite matrix  $Q \in \mathbf{R}^{\ell \times \ell}$ , the semi-norm associated with  $Q$  is denoted by  $\| \cdot \|_Q$ . In particular, for any vector  $x$ ,  $\|x\|_Q = \sqrt{x^T Q x}$ .

Throughout this paper, we make the following standard assumptions:

### Assumption 1

- (i)  $J$  is a convex, l.s.c function (not necessary differentiable) such that  $\text{dom} J \cup U \neq \emptyset$ .
- (ii)  $G$  is a convex and differentiable with its derivative Lipschitz of constant  $B_G$ .
- (iii)  $G + J$  is coercive on  $U$ , if  $U$  is not bounded, that is

$$\forall \{u^k | k \in N\} \subset U, \lim_{k \rightarrow +\infty} \|u^k\| = +\infty \Rightarrow \lim_{k \rightarrow +\infty} (G + J)(u^k) = +\infty.$$

- (iv)

$$0 \in \text{interior of } (A(U) - b). \tag{5}$$

The Lagrangian function of problem (1) is given by

$$L(u, p) = (G + J)(u) + \langle p, Au - b \rangle.$$

The pair  $(u^*, p^*) \in U \times R^m$  is called a saddle point of Lagrangian function  $L(u, p)$  if it holds that

$$L(u^*, p) \leq L(u^*, p^*) \leq L(u, p^*), \quad \forall u \in U, \quad \forall p \in \mathbf{R}^m. \quad (6)$$

Note that (iii) and (iv) in Assumption 1 are imposed to guarantee the existence of saddle point of  $L$  on  $U \times \mathbf{R}^m$ . Moreover, under Assumption 1,  $L$  and  $L_\gamma$  have the same sets of saddle points. The interested readers are referred to the book of Bertsekas [5] and the paper of Cohen and Zhu [14] for more results on Lagrangian duality theory. In addition to Assumption 1, the desired theoretical properties of VAPP, some structures of the core function is required as well.

**Assumption 2**  $K^k$  are strongly convex with parameter  $\beta^k$  and differentiable with its gradient Lipschitz continuous with parameter  $B^k$  on  $U$ . Moreover, there exist positive numbers  $\beta$  and  $B$  such that  $0 < \beta \leq \beta^k \leq B^k \leq B$ , for any  $k \in \mathbb{N}$ .

### 3 Convergence Analysis

We first present the main theorem of this paper, which states that the sequence generated by VAPP is actually convergent under some mild conditions.

**Theorem 1** Suppose Assumption 1 and Assumption 2 hold. Then by picking

$$0 < \epsilon < \beta^k / (B_G + \gamma \cdot \lambda_{\max}(A^\top A)) \quad \text{and} \quad 0 < \rho < (2 - \delta)\gamma, \quad (7)$$

the sequence  $\{(u^k, p^k)\}$  generated by VAPP converges to  $(u^*, p^*)$ , which is the saddle point of  $L$  over  $U \times R^m$ .

We would like to remark that the standard convergence analysis of ADMM works only when the coupled term in the objective of problem (1) disappears, i.e.,  $G(u) = 0$ . However, when the variables  $u_1, \dots, u_N$  is no longer separated in the objective, our algorithm still works and the convergence is guaranteed.

The rest of this section is dedicated to the proof of Theorem 1. To this end, we first present some lemmas as preparation.

**Lemma 1** Suppose  $K^k$  satisfies Assumption 2, then

(i) it holds that

$$K^k(u) - K^k(v) - \langle \nabla K^k, u - v \rangle \geq \frac{\beta^k}{2} \|u - v\|^2 \geq \frac{\beta}{2} \|u - v\|^2, \quad \forall u, v \in U, \quad (8)$$

$$K^k(u) - K^k(v) - \langle \nabla K^k, u - v \rangle \leq \frac{B^k}{2} \|u - v\|^2 \leq \frac{B}{2} \|u - v\|^2, \quad \forall u, v \in U. \quad (9)$$

(ii) Moreover, when the sequence  $\{m^k\}$  is constructed such that

$$1 \leq m_k \leq (\beta/B)m_{k+1} \leq m_{k+1} \leq \frac{1}{1-\delta}, \quad (10)$$

with some  $0 \leq \delta < 1$ , then for all  $u, v \in U$ , one has

$$K^{k+1}(u) - K^{k+1}(v) - \langle \nabla K^{k+1}(v), u - v \rangle \leq \frac{m_{k+1}}{m_k} \left( K^k(u) - K^k(v) - \langle \nabla K^k(v), u - v \rangle \right). \quad (11)$$

*Proof.* The results in (i) are standard. See for example Cohen [13], and Descent Lemma in [3, 14]. Now let's prove (ii). Suppose sequence  $\{m_k\}$  satisfies (10), then for all  $u, v \in U$  we have

$$\begin{aligned} K^{k+1}(u) - K^{k+1}(v) - \langle \nabla K^{k+1}, u - v \rangle &\leq \frac{B}{2} \|u - v\|^2 = \frac{B}{\beta^k} \frac{\beta^k}{2} \|u - v\|^2 \\ &\leq \frac{B}{\beta^k} \left( K^k(u) - K^k(v) - \langle \nabla K^k, u - v \rangle \right) \\ &\leq \frac{B}{\beta} \left( K^k(u) - K^k(v) - \langle \nabla K^k, u - v \rangle \right) \\ &\leq \frac{m_{k+1}}{m_k} \left( K^k(u) - K^k(v) - \langle \nabla K^k, u - v \rangle \right). \end{aligned}$$

□

**Lemma 2** *If function  $f$  is a convex and differentiable with its derivative Lipschitz of constant  $B_f$  on convex set  $U$ , then for  $\forall u, v, w \in U$  we have*

$$\langle \nabla f(u), w - v \rangle \leq f(w) - f(v) + \frac{B_f}{2} \|u - v\|^2.$$

*Proof.* From the convexity of  $f$  and (9), we have

$$\begin{aligned} \langle \nabla f(u), w - v \rangle &= \langle \nabla f(u), w - u \rangle + \langle \nabla f(u), u - v \rangle \\ &\leq [f(w) - f(u)] + [f(u) - f(v) + \frac{B_f}{2} \|u - v\|^2] \\ &= f(w) - f(v) + \frac{B_f}{2} \|u - v\|^2. \end{aligned}$$

□

**Lemma 3** *Suppose Assumption 1 and Assumption 2 hold,  $(u^*, p^*)$  is any saddle point of  $L$ , and  $(u^k, p^k)$  is generated by VAPP. Then it holds that*

$$\begin{aligned} &\langle \nabla K^k(u^k) - \nabla K^k(u^{k+1}), u^* - u^{k+1} \rangle + \frac{\epsilon}{2\rho} \left( \|p^{k+1} - p^*\|^2 - \|p^k - p^*\|^2 \right) \\ &\leq \frac{\epsilon}{2} \left( B_G \|u^k - u^{k+1}\|^2 + (\rho - \gamma) \|Au^{k+1} - b\|^2 - \gamma \|Au^k - b\|^2 + \gamma \|A(u^{k+1} - u^k)\|^2 \right) \quad (12) \end{aligned}$$



*Proof.* Recall that in every iteration of VAPP the subproblem is given by

$$u^{k+1} = \arg \min_{u \in U} K^k(u) - \langle \nabla K^k(u^k), u \rangle + \epsilon [\langle \nabla G(u^k), u \rangle + J(u) + \langle p^k + \gamma(Au^k - b), Au \rangle] \quad (13)$$

Since  $K^k(\cdot)$  is strongly convex, the unique solution  $u^{k+1}$  of is characterized by the following variational inequality:

$$\begin{aligned} \langle \nabla K^k(u^{k+1}) - \nabla K^k(u^k), u - u^{k+1} \rangle + \epsilon \left( \langle \nabla G(u^k), u - u^{k+1} \rangle + J(u) - J(u^{k+1}) \right. \\ \left. + \langle p^k + \gamma(Au^k - b), A(u - u^{k+1}) \rangle \right) \geq 0 \quad \forall u \in U. \end{aligned} \quad (14)$$

By taking  $u = u^*$ , one has that

$$\begin{aligned} & \langle \nabla K^k(u^k) - \nabla K^k(u^{k+1}), u^* - u^{k+1} \rangle \\ & \leq \epsilon \left( \langle \nabla G(u^k), u^* - u^{k+1} \rangle + J(u^*) - J(u^{k+1}) + \langle p^k + \gamma(Au^k - b), A(u^* - u^{k+1}) \rangle \right) \\ & \leq \epsilon \left( (G + J)(u^*) - (G + J)(u^{k+1}) + \frac{B_G}{2} \|u^k - u^{k+1}\|^2 + \langle p^k + \gamma(Au^k - b), A(u^* - u^{k+1}) \rangle \right) \end{aligned} \quad (15)$$

where the second last inequality is due to Lemma 2. Furthermore, since  $(u^*, p^*)$  is a saddle point,

$$\begin{aligned} (G + J)(u^*) &= (G + J)(u^*) + \langle p^*, Au^* - b \rangle \\ &= L(u^*, p^*) \\ &\leq L(u^{k+1}, p^*) = (G + J)(u^{k+1}) + \langle p^*, Au^{k+1} - b \rangle \end{aligned} \quad (16)$$

Combining (15), (16) and the fact  $Au^* = b$  yields

$$\begin{aligned} & \langle \nabla K^k(u^k) - \nabla K^k(u^{k+1}), u^* - u^{k+1} \rangle \\ & \leq \epsilon \left( \frac{B_G}{2} \|u^k - u^{k+1}\|^2 + \langle p^* - p^k - \gamma(Au^k - b), Au^{k+1} - b \rangle \right) \end{aligned} \quad (17)$$

On the other hand, from the dual update in VAPP

$$p^{k+1} = p^k + \rho \left( \sum_{i=1}^N A_i u_i^{k+1} - b \right), \quad (18)$$

it follows that

$$\begin{aligned} & \|p^{k+1} - p^*\|^2 \\ &= \|p^k - p^*\|^2 + \rho^2 \|Au^{k+1} - b\|^2 + 2\rho \langle p^k - p^*, Au^{k+1} - b \rangle \\ &= \|p^k - p^*\|^2 + \rho(\rho - 2\gamma) \|Au^{k+1} - b\|^2 + 2\rho \langle p^k - p^* + \gamma(Au^{k+1} - b), Au^{k+1} - b \rangle. \end{aligned}$$

As a result,

$$\begin{aligned} & \frac{\epsilon}{2\rho} \|p^{k+1} - p^*\|^2 - \frac{\epsilon}{2\rho} \|p^k - p^*\|^2 \\ &= \epsilon \left( \frac{(\rho - 2\gamma)}{2} \|Au^{k+1} - b\|^2 + \langle p^k - p^* + \gamma(Au^{k+1} - b), Au^{k+1} - b \rangle \right). \end{aligned} \quad (19)$$

Adding (19) and (17) together yields that

$$\begin{aligned} & \langle \nabla K^k(u^k) - \nabla K^k(u^{k+1}), u^* - u^{k+1} \rangle + \frac{\epsilon}{2\rho} \left( \|p^{k+1} - p^*\|^2 - \|p^k - p^*\|^2 \right) \\ & \leq \frac{\epsilon}{2} \left( B_G \|u^k - u^{k+1}\|^2 + (\rho - \gamma) \|Au^{k+1} - b\|^2 - \gamma \|Au^{k+1} - b\|^2 + 2\gamma \langle A(u^{k+1} - u^k), Au^{k+1} - b \rangle \right) \\ & = \frac{\epsilon}{2} \left( B_G \|u^k - u^{k+1}\|^2 + (\rho - \gamma) \|Au^{k+1} - b\|^2 - \gamma \|Au^k - b\|^2 + \gamma \|A(u^{k+1} - u^k)\|^2 \right). \end{aligned}$$

□

Now we are ready to prove the convergence of VAPP.

### Proof of Theorem 1

*Proof.* Suppose sequence  $\{m_k\}$  is chosen according to (10). First we define the following function

$$\Lambda^k(u, p) = \frac{1}{m_k} \left( K^k(u^*) - K^k(u) - \langle \nabla K^k(u), u^* - u \rangle + \frac{\epsilon}{2\rho} \|p - p^*\|^2 - \frac{\gamma\epsilon}{2} \|Au - b\|^2 \right), \quad (20)$$

where  $(u^*, p^*)$  is a saddle point associated with the Lagrangian function  $L(u, p)$ . Then due to the strongly convexity of  $K^k$ , we have that

$$\begin{aligned} \Lambda^k(u, p) &= \frac{1}{m_k} \left( K^k(u^*) - K^k(u) - \langle \nabla K^k(u), u^* - u \rangle + \frac{\epsilon}{2\rho} \|p - p^*\|^2 - \frac{\gamma\epsilon}{2} \|Au - Au^*\|^2 \right) \\ &\geq \frac{1}{m_k} \left( \frac{\beta^k}{2} \|u^* - u\|^2 + \frac{\epsilon}{2\rho} \|p - p^*\|^2 - \frac{\gamma\epsilon}{2} \lambda_{\max}(A^\top A) \|u - u^*\|^2 \right) \\ &\geq \frac{1}{m_k} \left( \frac{1}{2} \left( \beta^k - \gamma\epsilon \lambda_{\max}(A^\top A) \right) \|u - u^*\|^2 + \frac{\epsilon}{2\rho} \|p - p^*\|^2 \right) \geq 0, \end{aligned} \quad (21)$$

where the last inequality follows from the choice of  $\epsilon$  and  $\rho$  in (7). That is the distance between  $(u, p)$  and saddle point  $(u^*, p^*)$  is quantified by  $\Lambda^k(u, p)$ . As a result, to achieve the convergence, it suffices to study the sequence  $\{\Lambda^k(u^k, p^k)\}$ .

According to Assumption 2 and Lemma 1, one has that

$$\begin{aligned}
& \Lambda^{k+1}(u^{k+1}, p^{k+1}) - \Lambda^k(u^k, p^k) \\
&= \frac{1}{m_{k+1}} \left( K^{k+1}(u^*) - K^{k+1}(u^{k+1}) - \langle \nabla K^{k+1}(u^{k+1}), u^* - u^{k+1} \rangle \right) \\
&\quad - \frac{1}{m_k} \left( K^k(u^*) - K^k(u^k) - \langle \nabla K^k(u^k), u^* - u^k \rangle \right) \\
&\quad + \frac{\epsilon}{2\rho m_{k+1}} \|p^{k+1} - p^*\|^2 - \frac{\gamma\epsilon}{2m_{k+1}} \|Au^{k+1} - b\|^2 - \frac{\epsilon}{2\rho m_k} \|p^k - p^*\|^2 + \frac{\gamma\epsilon}{2m_k} \|Au^k - b\|^2 \\
&\leq \frac{1}{m_k} \left( K^k(u^*) - K^k(u^{k+1}) - \langle \nabla K^k(u^{k+1}), u^* - u^{k+1} \rangle - K^k(u^*) + K^k(u^k) + \langle \nabla K^k(u^k), u^* - u^k \rangle \right) \\
&\quad + \frac{\epsilon}{2\rho m_k} \|p^{k+1} - p^*\|^2 - \frac{\epsilon}{2\rho m_k} \|p^k - p^*\|^2 - \frac{\gamma\epsilon}{2m_{k+1}} \|Au^{k+1} - b\|^2 + \frac{\gamma\epsilon}{2m_k} \|Au^k - b\|^2 \\
&= \frac{1}{m_k} \left( K^k(u^k) - K^k(u^{k+1}) - \langle \nabla K^k(u^k), u^k - u^{k+1} \rangle \right) \tag{22}
\end{aligned}$$

$$\begin{aligned}
& + \frac{1}{m_k} \left( \langle \nabla K^k(u^k) - \nabla K^k(u^{k+1}), u^* - u^{k+1} \rangle + \frac{\epsilon}{2\rho} (\|p^{k+1} - p^*\|^2 - \|p^k - p^*\|^2) \right) \tag{23} \\
& - \frac{\gamma\epsilon}{2m_{k+1}} \|Au^{k+1} - b\|^2 + \frac{\gamma\epsilon}{2m_k} \|Au^k - b\|^2.
\end{aligned}$$

The convexity of  $K^k(\cdot)$  implies that formula (22) is less than  $-\frac{\beta^k}{2m_k} \|u^k - u^{k+1}\|^2$ . Note that (23) can further be bounded above by using Lemma 3. Therefore,

$$\begin{aligned}
& \Lambda^{k+1}(u^{k+1}, p^{k+1}) - \Lambda^k(u^k, p^k) \\
&\leq \frac{1}{m_k} \left( -\frac{(\beta^k - B_G\epsilon)}{2} \|u^k - u^{k+1}\|^2 + \frac{\epsilon}{2} \left( (\rho - (1 + \frac{m_k}{m_{k+1}})\gamma) \|Au^{k+1} - b\|^2 + \gamma \|A(u^{k+1} - u^k)\|^2 \right) \right) \\
&\leq \frac{1}{m_k} \left( -\frac{(\beta^k - B_G\epsilon)}{2} \|u^k - u^{k+1}\|^2 + \frac{\gamma\epsilon}{2} \lambda_{\max}(A^\top A) \|u^{k+1} - u^k\|^2 + \frac{\epsilon}{2} (\rho - (2 - \delta)\gamma) \|Au^{k+1} - b\|^2 \right) \\
&\leq \frac{1}{m_k} \left( \frac{1}{2} \left( \epsilon (B_G + \gamma \lambda_{\max}(A^\top A)) - \beta^k \right) \|u^k - u^{k+1}\|^2 + \frac{\epsilon}{2} (\rho - (2 - \delta)\gamma) \|Au^{k+1} - b\|^2 \right) \leq 0. \tag{24}
\end{aligned}$$

where the second inequality follows as  $m_{k+1} \leq 1/(1 - \delta) \leq m_k/(1 - \delta)$  and the last inequality is due to the choice of  $\epsilon$  and  $\rho$  defined in (7). Consequently  $\{\Lambda^k(u^k, p^k)\}$  is nonincreasing. This combined with (21) implies that  $\{\Lambda^k(u^k, p^k)\}$  has a limit,

$$\lim_{k \rightarrow \infty} \|p^{k+1} - p^k\|/\rho = \lim_{k \rightarrow \infty} \|Au^{k+1} - b\| = 0 \quad \text{and} \quad \lim_{k \rightarrow \infty} \|u^{k+1} - u^k\| = 0 \tag{25}$$

Moreover, (21) and boundedness of  $\{\Lambda^k(u^k, p^k)\}$  implies that  $\{u^k\}$  and  $\{p^k\}$  are bounded as well. Therefore the sequence  $\{(u^k, p^k)\}$  has a cluster point  $(\bar{u}, \bar{p})$ . Taking the limit in (25) gives that

$$A\bar{u} - b = 0. \tag{26}$$

Furthermore, since gradients of  $K^k$  and  $G$  are Lipschitz continuous,

$$\lim_{k \rightarrow \infty} \|\nabla K^k(u^{k+1}) - \nabla K^k(u^k)\| = 0 \quad \text{and} \quad \lim_{k \rightarrow \infty} \|\nabla G(u^{k+1}) - \nabla G(u^k)\| = 0.$$

Now letting  $k+1 \rightarrow \infty$  in (14) and then combining the formulas above together with the convexity of  $G$  and (25) yields

$$(G+J)(u) - (G+J)(\bar{u}) + \langle \bar{p}, Au - b \rangle \geq 0, \forall u \in U. \quad (27)$$

Consequently (26) and (27) imply that

$$L(\bar{u}, p) = L(\bar{u}, \bar{p}) = (G+J)(\bar{u}) + \langle p, A\bar{u} - b \rangle \leq (G+J)(u) + \langle \bar{p}, Au - b \rangle = L(u, \bar{p}), \quad \forall p \in R^m, \quad \forall u \in U.$$

From the definition of saddle point, it holds that  $(\bar{u}, \bar{p}) \in U^* \times P^*$ .

Note that the argument above goes through as long as  $(u^*, p^*)$  is a saddle point of Lagrangian function  $L(u, p)$ . Therefore, we can set  $u^* = \bar{u}$ ,  $p^* = \bar{p}$  and taking limit for sequence  $\{\Lambda^k(u^k, p^k)\}$ . From construction (20) of  $\Lambda^k(u, v)$ , we know that zero is a cluster point of  $\{\Lambda^k(u^k, p^k)\}$ . Moreover, we have shown that the limit of the total sequence  $\{\Lambda^k(u^k, p^k)\}$  exists. As a result,  $\Lambda^k(u^k, p^k) \rightarrow 0$ . This combined with (21) implies that  $u^k \rightarrow u^*$  and  $p^k \rightarrow p^*$ .

□

At the end of this section, we discuss some choices of core functions that satisfy the conditions in Theorem 1, and thus guarantee the convergence.

First, let's take

$$K^k(u) = \sum_{i=1}^N \nu_i G(u_1^k, \dots, u_{i-1}^k, u_i, u_{i+1}^k, \dots, u_n^k) + \sum_{i=1}^N \frac{\theta_i}{2} \|A_i u_i + (\sum_{j \neq i} A_j u_j^k - b)\|^2 + \sum_{i=1}^N \frac{\alpha_i}{2} \|P_i u_i\|^2 \quad (28)$$

in problem (1), where  $\nu_i > 0$ ,  $\theta_i \geq 0$  and  $\alpha_i \geq 0$  for all  $i = 1, \dots, N$ . Therefore,  $\beta^k \geq B_G + \gamma \cdot \lambda_{\max}(A^\top A)$  suffices to enable the sequence generated by VAPP convergent. When  $\nu_i = 0$ ,  $B_G$  is no longer useful.

The second choice is mentioned in the introduction part. Specifically, we let  $K^k(u) = \frac{1}{2}(u - u^k)^\top D(u^k)(u - u^k)$ , where  $D(u^k)$  is an  $n \times n$  matrix. In the following, we specify some choices of  $D(u^k)$  that leads to a few Newton-like methods [28, 46]:

- (1) *Newton's method*:  $D(u^k) = \nabla^2 G(u^k)$ .
- (2) *Quasi-Newton methods*:  $D(u^k) \approx \nabla^2 G(u^k)$ .
- (3) *Linearized Jaccobi method*:  $D(u^k) = M(u^k)$ , where  $M(u^k)$  is the diagonal part of  $\nabla^2 G(u^k)$ .

- (4) *SOR method*:  $D(u^k) = N(u^k) + M(u^k)/\omega$ , where  $N(u^k)$  is the lower triangular part of  $\nabla^2 G(u^k)$  and  $\omega \in (0, 2)$ .

If the Hessian matrix  $\nabla^2 G(u^k)$  is positive definite on  $U$ , the convergence of Newton's method is assured.

## 4 Adaptively Tuning Strategy on Parameter $\epsilon$

By reinvestigating the proof of Theorem 1, we know the reason of specifying  $\epsilon$  in (7) is to ensure the monotonicity of sequence  $\{\Lambda^k(u^k, p^k)\}$  defined in (20). A consequence of condition (7) is that the value of  $\epsilon$  could be very small, which in turn could slow down the convergence. To alleviate such drawback, we define

$$\begin{aligned}\Delta(u^k, u^{k+1}) &= -\frac{(\beta^k - B_G \epsilon)}{2} \|u^k - u^{k+1}\|^2 + \frac{\epsilon \gamma}{2} \|A(u^{k+1} - u^k)\|^2 \\ &\leq \frac{1}{2} \left( \epsilon \left( B_G + \gamma \lambda_{\max}(A^\top A) \right) - \beta^k \right) \|u^k - u^{k+1}\|^2.\end{aligned}$$

Therefore, for a small positive scalar  $\eta$ , when  $\epsilon$  is small enough, we must have

$$\Delta(u^k, u^{k+1}) \leq -\eta \|u^k - u^{k+1}\|^2. \quad (29)$$

Once the above inequality is satisfied, according to the second line of formula (24), one has that

$$\begin{aligned}&\Lambda^{k+1}(u^{k+1}, p^{k+1}) - \Lambda^k(u^k, p^k) \\ &\leq \frac{1}{m_k} \left( \Delta(u^k, u^{k+1}) + \frac{\epsilon}{2} (\rho - (2 - \delta)\gamma) \|Au^{k+1} - b\|^2 \right) \\ &\leq \frac{1}{m_k} \left( -\eta \|u^k - u^{k+1}\|^2 + \frac{\epsilon}{2} (\rho - (2 - \delta)\gamma) \|Au^{k+1} - b\|^2 \right).\end{aligned}$$

This combined with the choice of  $\rho$  given in (7) guarantees the sequence  $\{\Lambda^k(u^k, p^k)\}$  is monotonically decreasing and the convergence of  $\{(u^k, p^k)\}$ . Therefore, similar to [17], we propose an adaptively parameter tuning strategy. The difference is that the tuned parameter here is a scalar  $\epsilon$  instead of a matrix. Based on the values of  $\Delta(u^k, u^{k+1})$ , we proposed the VAPP with Adaptive Parameter Tuning as follows:

---

### VAPP with Adaptive Parameter Tuning (VAPP-APT)

---

Initialize with a large  $\epsilon^0$ , a small  $\eta$ ,  $u^0 \in U$  and  $p^0$ ;

**set**  $k = 1$ , the error tolerance  $\nu$

**while**  $\|u^{k-1} - u^k\| > \nu$ , **do**

$$u^{k+1} = \arg \min_{u \in U} K^k(u) - \langle \nabla K^k(u^k), u \rangle + \epsilon^k [\langle \nabla G(u^k), u \rangle + J(u) + \langle p^k + \gamma(Au^k - b), Au \rangle] ;$$

$$p^{k+1} = p^k + \rho(\sum_{i=1}^N A_i u_i^{k+1} - b);$$

**if**  $\Delta(u^k, u^{k+1}) < -\eta \|u^k - u^{k+1}\|^2$ , **then**

$$\epsilon^{k+1} \leftarrow \epsilon^k;$$

$$k \leftarrow k + 1;$$

**else**

$$\text{Decrease } \epsilon^k: \epsilon^k \leftarrow \mu \epsilon^k + \tau \text{ } (0 < \mu < 1, \tau \leq 0);$$

**end if**

**end while**

---

In the above algorithm, we start with a reasonable large  $\epsilon$  and gradually decrease its value. Due to Theorem 1, when the parameter  $\epsilon$  is small enough such that condition (7) holds, the inequality (29) will be satisfied. Therefore, the adjustment of  $\epsilon$  can only occur finite many times, then  $\epsilon$  will remain constant. Consequently, the convergence of incorporating VAPP with such adaptively tuning strategy follows immediately from our discussion above.

**Theorem 2** *The sequence  $\{(u^k, p^k)\}$  generated by (VAPP-APT) converges to the saddle point of  $L$  over  $U \times R^m$ .*

Similar to the case reported in [17], in our numerical experiments we find that the parameter  $\epsilon$  typically adjust itself only at the very beginning of the algorithm and then remain constant afterwards, moreover the the resulting value of  $\epsilon$  is usually much larger than that required by the condition (7), which yields a significant faster convergence in practice.

## 5 Convergence Rate Analysis with Quadratic Core Function

In this section, we assume  $\epsilon$  meets the condition (7) and the core function is endowed with the quadratic form (4):

$$K^k(u) = \sum_{i=1}^N \frac{\theta_i}{2} \|A_i u_i + (\sum_{j \neq i} A_j u_j^k - b)\|^2 + \sum_{i=1}^N \frac{\alpha_i}{2} \|P_i u_i\|^2,$$

where  $\theta_i \geq 0$  and  $\alpha_i \geq 0$  for all  $i = 1, \dots, N$ . In our later numerical implementations, we consider three specializations of (4) corresponding to three different algorithms:

- (1)  $\theta_i > 0$  and  $\alpha_i > 0$  for all  $i = 1, \dots, N$ , and we call it Proximal Jacobian Varying Auxiliary Problem Principle (PJVAPP);
- (2)  $\theta_i > 0$  and  $\alpha_i = 0$  for all  $i = 1, \dots, N$ , that is  $K^k(u) = \sum_{i=1}^N \frac{\theta_i}{2} \|A_i u_i + \sum_{j \neq i} A_j u_j^k - b\|^2$ . We name this algorithm Jacobian Varying Auxiliary Problem Principle (JVAPP);
- (3)  $\theta_i = 0$  and  $\alpha_i > 0$  for all  $i = 1, \dots, N$ , that is  $K^k(u) = \sum_{i=1}^N \frac{\alpha_i}{2} \|P_i u_i\|^2$ . We call this algorithm Proximal Varying Auxiliary Problem Principle (PVAPP).

We remark that the various Jacobian type ADMM, ( including PJADM, prox-linear JADM, JADM) considered in [17] are specifications or closely related to the above VAPP type algorithms. To be clear, we present their specific iteration schemes in the following.

---

JADM (or PJADM)

---

Initialize  $u^0 \in U$  and  $p^0$

**for**  $k = 0, 1, \dots$ , **do**

**for**  $i = 1, \dots, N$ , **do**

$$u_i^{k+1} = \arg \min_{u \in U_i} J_i(u_i) + \langle p^k, A_i u_i - b \rangle + \frac{\gamma}{2} \|A_i u_i + \sum_{j \neq i} A_j u_j^k - b\|^2 ;$$

$$(\text{or } u_i^{k+1} = \arg \min_{u \in U_i} J_i(u_i) + \langle p^k, A_i u_i - b \rangle + \frac{\gamma}{2} \|A_i u_i + \sum_{j \neq i} A_j u_j^k - b\|^2 + \frac{1}{2} \|u_i - u_i^k\|_{\bar{P}_i}^2 \text{ for PJADM})$$

**end for**

$$p^{k+1} = p^k + \rho(Au^{k+1} - b).$$

**end for**

---

In particular, by choosing  $\epsilon = 1$ ,  $\theta_i = \gamma = \rho$ ,  $\alpha_i = 1$  and  $\bar{P}_i = P_i^\top P_i$  for all  $i = 1, \dots, N$ , PJVAPP is actually the PJADM. Besides, PVAPP is the Algorithm 14 considered in [14] and this algorithm further reduces to the so-called prox-linear JADM when  $\epsilon = 1$ ,  $\gamma = \rho$  and  $P_i = I$  for all  $i = 1, \dots, N$ . Moreover, JADM and JVAPP with  $\theta_i = \gamma = \rho$  are almost the same except the the extra linear term  $(\gamma - \frac{\theta_i}{\epsilon}) \langle Au^k - b, A_i u_i \rangle$  in the subproblem of JVAPP.

Our goal in this section is to analyze the rate of convergence with the above quadratic core function. First, we show that Theorem 1 holds in this case with some mild conditions, and thus the convergence immediately follows.

**Proposition 1** *When the core function is given by (4), Assumption 2 is automatically satisfied. Moreover, suppose for each index  $i = 1, \dots, N$ , we either have  $A_i$  has full column rank with  $\theta_i > 0$  or  $P_i$  has full column rank with  $\alpha_i > 0$ , then the convergence of VAPP is guaranteed.*

*Proof.* Denoting  $c_i^k = \sum_{j \neq i} A_j u_j^k - b \ \forall i = 1, 2, \dots, N$ , we first calculate the gradient of the core function:

$$\nabla(K^k)(u) = \begin{pmatrix} \theta_1 A_1^T (A_1 u_1 + c_1^k) + \alpha_1 P_1^T P_1 u_1 \\ \vdots \\ \theta_N A_N^T (A_N u_N + c_N^k) + \alpha_N P_N^T P_N u_N \end{pmatrix}$$

It suffices to show Assumption 2 is satisfied under this setting. First, it is not hard to verify that

$$\begin{aligned} K^k(u) - K^k(v) - \langle \nabla K^k(v), u - v \rangle &= \sum_{i=1}^N \left( \frac{\theta_i}{2} \|A_i(u_i - v_i)\|^2 + \frac{\alpha_i}{2} \|P_i(u_i - v_i)\|^2 \right) \\ &\geq \left( \min_i \frac{\theta_i}{2} \lambda_{\min}(A_i^T A_i) + \min_i \frac{\alpha_i}{2} \lambda_{\min}(P_i^T P_i) \right) \cdot \|u - v\|^2 \end{aligned}$$

Therefore, our conditions on matrices  $A_i$ s and  $P_i$ s imply that  $K^k$  is strongly convex with constant  $\beta = \beta^k = \min_i \{\theta_i \lambda_{\min}(A_i^T A_i)\} + \min_i \{\alpha_i \lambda_{\min}(P_i^T P_i)\}$ , for all  $k$ . Next, a direct computation impliest that

$$\nabla(K^k)(u) - \nabla(K^k)(v) = \begin{pmatrix} (\theta_1 A_1^T A_1 + \alpha_1 P_1^T P_1) (u_1 - v_1) \\ \vdots \\ (\theta_N A_N^T A_N + \alpha_N P_N^T P_N) (u_N - v_N) \end{pmatrix}.$$

Consequently  $\|\nabla(K^k)(u) - \nabla(K^k)(v)\| \leq \max_i (\theta_i \lambda_{\max}(A_i^T A_i) + \alpha_i \lambda_{\max}(P_i^T P_i)) \cdot \|u - v\|$ . That the gradient of  $K^k$  is Lipchitz continuous with constant  $B = B_k = \max_i (\theta_i \lambda_{\max}(A_i^T A_i) + \alpha_i \lambda_{\max}(P_i^T P_i))$ , for all  $k$ . In this case, we can take  $m_{k+1} = m_k = 1$  and  $\delta = 0$  in (ii) of Lemma 1.  $\square$

Note that in the absence of  $G$ , when  $A_i$  has full column rank for all  $i$ , the above results guarantee the convergence of PJVAPP, and this condition is weaker than that of JADM in [17], where  $A_i$ s are additional required to be mutually near-orthogonal.

To study the convergence rate, following [17] we adopt the distance of the two points from the two consecutive iterations to measure the optimality of the solution at current step. To facilitate our discussion, we suppose  $G(\cdot)$  is twice differentiable and let

$$H := \frac{1}{\epsilon} \begin{pmatrix} \theta_1 A_1^T A_1 + \alpha_1 P_1^T P_1 & & & \\ & \ddots & & \\ & & \theta_i A_i^T A_i + \alpha_i P_i^T P_i & \\ & & & \ddots \\ & & & & \theta_N A_N^T A_N + \alpha_N P_N^T P_N \end{pmatrix}$$

$$Q^k := \int_0^1 \nabla^2 G(u^k + \tau(u^{k-1} - u^k)) d\tau \quad \text{and} \quad \tilde{H}^k := H - \gamma A^T A - Q^k$$



We assume the hessian matrix of  $G(\cdot)$  is bounded above, i.e., there exists some  $0 < M < \infty$  such that  $0 \preceq \nabla^2 G(u) \preceq MI$ ,  $\forall u \in U$ , which yields that  $0 \preceq Q^k \preceq MI$ . We denote

$$\underline{H} := H - \gamma A^T A - MI, \quad \overline{H} := H - \gamma A^T A. \quad (30)$$

Then, obviously  $\underline{H} \preceq \tilde{H}^k \preceq \overline{H}$  for all  $k$ .

To proceed, we quote a lemma in [17], which is useful to establish the  $o(1/k)$  convergence rate in this paper.

**Lemma 4** *If a sequence  $\{a_k\} \subseteq \mathbb{R}$  obeys: (1)  $a_k \geq 0$ ; (2)  $\sum_{k=1}^{\infty} a_k < +\infty$ ; (3)  $a_k$  is monotonically non-increasing, then we have  $a_k = o(1/k)$ .*

Inspired by the above lemma, the key of our analysis is the monotonicity of sequence  $\|u^k - u^{k+1}\|_{\overline{H}}^2 + \frac{1}{\rho} \|p^k - p^{k+1}\|^2$ , which is proved in the following lemma.

**Lemma 5** *Suppose the sequence  $\{u^k, p^k\}$  is generated by VAPP,  $G(\cdot)$  is twice differentiable convex function. If  $\underline{H} \succeq 0$  and  $0 < \rho < 2\gamma$ , then it holds that*

$$\|u^k - u^{k+1}\|_{\overline{H}}^2 + \frac{1}{\rho} \|p^k - p^{k+1}\|^2 \leq \|u^{k-1} - u^k\|_{\overline{H}}^2 + \frac{1}{\rho} \|p^{k-1} - p^k\|^2 \quad (31)$$

*Proof.* To simplify the notation, we let

$$\Delta u^{k+1} = u^k - u^{k+1} \text{ and } \Delta p^{k+1} = p^k - p^{k+1}.$$

Consequently,  $\Delta u_i^{k+1} = u_i^k - u_i^{k+1}$ ,  $i = 1, \dots, N$ . Recall that the subproblem of VAPP is given by (13) and from the optimality condition it follows that for  $i = 1, \dots, N$  there exists a point  $y_i^{k+1} \in \partial J_i(u_i^{k+1})$  satisfying

$$\left( \frac{1}{\epsilon} \left( \nabla_{u_i} K^k(u^{k+1}) - \nabla_{u_i} K^k(u^k) \right) + \nabla_{u_i} G(u^k) + A_i^T \left( p^k + \gamma(Au^k - b) \right) + y_i^{k+1} \right)^T (u_i - u_i^{k+1}) \geq 0, \quad \forall u_i \in U_i.$$

Then for any  $i = 1, \dots, N$  and  $u_i \in U_i$ , plugging the gradient of the core function that defined in (4) into the above formula yields

$$\left( -\frac{\theta_i}{\epsilon} A_i^T A_i \Delta u_i^{k+1} - \frac{\alpha_i}{\epsilon} P_i^T P_i \Delta u_i^{k+1} + \nabla_{u_i} G(u^k) + A_i^T \left( p^k + \gamma(Au^k - b) \right) + y_i^{k+1} \right)^T (u_i - u_i^{k+1}) \geq 0. \quad (32)$$

Repeating the above argument for the  $k-1$ -th iteration gives that

$$\left( -\frac{\theta_i}{\epsilon} A_i^T A_i \Delta u_i^k - \frac{\alpha_i}{\epsilon} P_i^T P_i \Delta u_i^k + \nabla_{u_i} G(u^{k-1}) + A_i^T \left( p^{k-1} + \gamma(Au^{k-1} - b) \right) + y_i^k \right)^T (u_i - u_i^k) \geq 0. \quad (33)$$

Moreover, since  $J(\cdot)$  is convex, for any  $y_i^k \in \partial J_i(u_i^k)$  and  $y_i^{k+1} \in \partial J_i(u_i^{k+1})$

$$\langle y_i^k - y_i^{k+1}, u_i^k - u_i^{k+1} \rangle \geq 0.$$

Therefore, by taking  $u_i = u_i^k$  in (32) and  $u_i = u_i^{k+1}$  in (33), it holds that

$$\begin{aligned} 0 &\leq \sum_{i=1}^N \langle y_i^k - y_i^{k+1}, u_i^k - u_i^{k+1} \rangle \\ &\leq \sum_{i=1}^N \frac{\theta_i}{\epsilon} \langle A_i(\Delta u_i^k - \Delta u_i^{k+1}), A_i \Delta u_i^{k+1} \rangle + \sum_{i=1}^N \frac{\alpha_i}{\epsilon} \langle P_i(\Delta u_i^k - \Delta u_i^{k+1}), P_i \Delta u_i^{k+1} \rangle \\ &\quad - \langle \nabla G(u^{k-1}) - \nabla G(u^k), \Delta u^{k+1} \rangle - \langle \Delta p^k, A \Delta u^{k+1} \rangle - \gamma \langle A \Delta u^k, A \Delta u^{k+1} \rangle \\ &= -\|\Delta u^{k+1}\|_H^2 + (\Delta u^k)^T \tilde{H}^k \Delta u^{k+1} - \langle \Delta p^k, A \Delta u^{k+1} \rangle, \end{aligned}$$

where the equality follows as

$$\nabla G(u^{k-1}) - \nabla G(u^k) = \int_0^1 d \nabla G(u^k + \tau(u^{k-1} - u^k)) = \int_0^1 \nabla^2 G(u^k + \tau(u^{k-1} - u^k)) d\tau(u^{k-1} - u^k).$$

Since  $\overline{H} \succeq \tilde{H}^k \succeq \underline{H} \succeq 0$ , the above inequality implies that

$$\begin{aligned} 2\langle \Delta p^k, A \Delta u^{k+1} \rangle &\leq -2\|\Delta u^{k+1}\|_H^2 + 2(\Delta u^k)^T \tilde{H}^k \Delta u^{k+1} \\ &\leq -2\|\Delta u^{k+1}\|_H^2 + \|\Delta u^k\|_{\tilde{H}^k}^2 + \|\Delta u^{k+1}\|_{\tilde{H}^k}^2 \\ &\leq -2\|\Delta u^{k+1}\|_H^2 + \|\Delta u^k\|_{\underline{H}}^2 + \|\Delta u^{k+1}\|_{\underline{H}}^2 \end{aligned} \tag{34}$$

Observing  $\Delta p^{k+1} = \Delta p^k + \rho A \Delta u^{k+1}$  is the dual update of VAPP, it follows that

$$\begin{aligned} \frac{1}{\rho} \|\Delta p^{k+1}\|^2 - \frac{1}{\rho} \|\Delta p^k\|^2 &= 2\langle \Delta p^k, A \Delta u^{k+1} \rangle + \rho \|A \Delta u^{k+1}\|^2 \\ &\leq -2\|\Delta u^{k+1}\|_H^2 + \|\Delta u^k\|_{\underline{H}}^2 + \|\Delta u^{k+1}\|_{\underline{H}}^2 + \rho \|A \Delta u^{k+1}\|^2. \end{aligned}$$

As a result,

$$\begin{aligned} &\left( \|\Delta u^{k+1}\|_{\underline{H}}^2 + \frac{1}{\rho} \|\Delta p^{k+1}\|^2 \right) - \left( \|\Delta u^k\|_{\underline{H}}^2 + \frac{1}{\rho} \|\Delta p^k\|^2 \right) \\ &\leq -2\|\Delta u^{k+1}\|_H^2 + 2\|\Delta u^{k+1}\|_{\underline{H}}^2 + \rho \|A \Delta u^{k+1}\|^2 \\ &= -\|\Delta u^{k+1}\|_{(2\gamma-\rho)A^T A}^2 \\ &\leq 0, \end{aligned}$$

the last inequality is due to the choice of  $\rho$  in (7) and the proof is complete.  $\square$

With those preparations in hand, the convergence rate of our algorithm readily follows.

**Theorem 3** *Suppose Assumption 1 holds, the core function is defined in (4), and for each index  $i = 1, \dots, N$ , we either have  $A_i$  has full column rank with  $\theta_i > 0$  or  $P_i$  has full column rank with  $\alpha_i > 0$ . Moreover, assume that  $\underline{H} \succeq 0$  and  $G(\cdot)$  is convex, twice differentiable and its hessian matrix is bounded. Suppose the sequence  $\{u^k, p^k\}$  is generated by VAPP and the values of the parameters  $\rho$  and  $\epsilon$  are chosen based on (7). Then it follows that  $\|u^k - u^{k+1}\|_{\bar{H}}^2 = o(1/k)$  and  $\|p^k - p^{k+1}\|^2 = o(1/k)$ .*

*Proof.* Due to the previous discussion and the choice of  $\epsilon, \rho$ , Theorem 1 follows. Therefore,

$$\begin{aligned} & \Lambda^{k+1}(u^{k+1}, p^{k+1}) - \Lambda^k(u^k, p^k) \\ & \leq \frac{1}{2} \left( \epsilon \left( B_G + \gamma \lambda_{\max}(A^\top A) \right) - \beta^k \right) \|u^k - u^{k+1}\|^2 + \frac{\epsilon}{2} (\rho - 2\gamma) \|Au^{k+1} - b\|^2, \end{aligned}$$

and  $\lim_{k \rightarrow \infty} \Lambda^k(u^k, p^k) = 0$ . Since  $\frac{1}{2} \left( \epsilon \left( B_G + \gamma \lambda_{\max}(A^\top A) \right) - \beta^k \right) \leq 0$  and  $\frac{\epsilon}{2} (\rho - 2\gamma) \leq 0$ , there exist positive numbers  $\eta_1$  and  $\eta_2$  such that

$$\frac{1}{2} \left( \epsilon \left( B_G + \gamma \lambda_{\max}(A^\top A) \right) - \beta^k \right) I \preceq -\eta_1 \bar{H} \quad \text{and} \quad \frac{\epsilon}{2} (\rho - 2\gamma) I \leq -\frac{\eta_2}{\rho} I.$$

By letting  $\eta = \min\{\eta_1, \eta_2\}$ , one has that

$$\Lambda^{k+1}(u^{k+1}, p^{k+1}) - \Lambda^k(u^k, p^k) \leq -\eta \left( \|u^k - u^{k+1}\|_{\bar{H}}^2 + \frac{1}{\rho} \|p^k - p^{k+1}\|^2 \right) \quad (35)$$

Summing (35) over  $k$  and taking limit yields

$$\sum_{k=1}^{\infty} \left( \|u^k - u^{k+1}\|_{\bar{H}}^2 + \frac{1}{\rho} \|p^k - p^{k+1}\|^2 \right) \leq \frac{1}{\eta} \Lambda^0(u^0, p^0) < \infty.$$

Moreover Lemma 5 implies that the sequence  $\{\|u^k - u^{k+1}\|_{\bar{H}}^2 + \frac{1}{\rho} \|p^k - p^{k+1}\|^2\}$  is non-increasing. Finally, the conclusion follows by combining those results with Lemma 4.  $\square$

In terms of the above theorem, we remark that in the case  $G \equiv 0$ , the semidefinite condition  $\underline{H} \succeq 0$  is not necessary, i.e., it is satisfied automatically. To see this, we recall that

$$\beta^k = \min_i \theta_i \lambda_{\min}(A_i^\top A_i) + \min_i \alpha_i \lambda_{\min}(P_i^\top P_i)$$

when the core function is given in (4). Due to the definition of  $H$ , we have that

$$\lambda_{\min}(H) = \frac{1}{\epsilon} \left( \min_i \theta_i \lambda_{\min}(A_i^\top A_i) + \min_i \alpha_i \lambda_{\min}(P_i^\top P_i) \right) = \frac{\beta_k}{\epsilon}$$

Moreover, when  $G \equiv 0$ ,

$$\underline{H} = \tilde{H}^k = \overline{H} = H - \gamma A^T A.$$

Consequently,

$$\begin{aligned} \langle \underline{H}x, x \rangle &= \langle (H - \gamma A^T A)x, x \rangle \\ &\geq \left( \lambda_{\min}(H) - \gamma \lambda_{\max}(A^T A) \right) \|x\|^2 \\ &= \left( \frac{\beta^k}{\epsilon} - \gamma \lambda_{\max}(A^T A) \right) \|x\|^2 \geq 0, \end{aligned}$$

where the last inequality follows from the fact that  $\epsilon$  is chosen according to (7). Thus,  $\underline{H} \succeq 0$  and the  $o(1/k)$  convergence rate of VAPP follows. Finally, we want to mention that, comparing to the PJADM of [17], the convergence rate can be achieved without adding proximal terms to all block variable  $u_i$ . In other words, the proximal term is not needed for the  $i$ -th block when  $A_i$  has full column rank and  $\theta_i > 0$ .

## 6 Numerical Experiments

In this section, we justify the capability of our new framework VAPP by some numerical results. We are about to apply the three specifications of VAPP mentioned at the beginning of Section 5: (i) PJVAPP, (ii) JVAPP and (iii) PVAPP, to some concrete problems, and compare their performances with that of other approaches. All of our numerical experiments are run in MATLAB(R2011b) on a personal computer with an Intel Core i5-33370 CPUs (1.80GHz) and 4.00 GB of RAM.

### 6.1 Divergence Example of JADM

In [30], He, Hou and Yuan provided the following example

$$\begin{aligned} \min_x \quad & 0 \\ \text{s.t.} \quad & x_1 + x_2 = 0, \end{aligned} \tag{36}$$

where the divergence of JADM was observed. Obvious, the above problem is a two block case of (1) with  $G(u) = J_i(u) = 0$ ,  $U_i = \mathbf{R}$ ,  $A = (A_1, A_2) = (1, 1)$ ,  $b = 0$ . We apply the three variants of VAPP to this example starting from the points  $(u_1^0, u_2^0)^\top = (1, 1)^\top$  and multiplier  $p^0 = 1$ . The convergence curves are plotted in Figure 1 and it shows that all of our three algorithms are convergent within 5 steps.

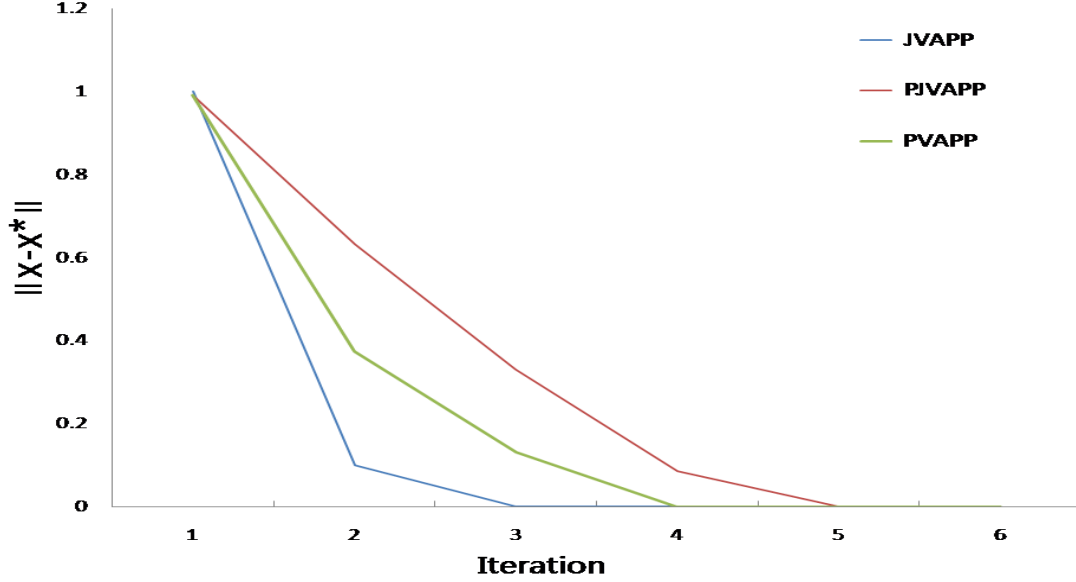


Figure 1: Convergence of VAPP by the problem (36)

## 6.2 Basis Pursuit Problem

In this subsection, we test the efficiency of our approach for the following *basis pursuit* problem:

$$\begin{aligned} \min_x \quad & \|x\|_1 \\ \text{s.t.} \quad & Ax = c, \end{aligned} \tag{37}$$

where  $x \in \mathbf{R}^n$ ,  $A \in \mathbf{R}^{m \times n}$  and  $c \in \mathbf{R}^m$  ( $m < n$ ). This problem aims to find a sparse solution of a linear system, and has wide applications in compressive sensing, signal and image processing, statistics, and et. al.. In the case that the data set has  $N$ -block structure:  $x = [x_1, x_2, \dots, x_N]$  and  $A = [A_1, A_2, \dots, A_N]$ . Then this problem is in the form of (1) with  $G(u) = 0$  and  $J_i(u_i) = \|x_i\|$ .

In our experiment, we randomly construct an instance of such problem as follows. First a sparse solution  $x^*$  is randomly generated with  $k$  ( $k \ll n$ ) nonzeros, which are drawn from the standard normal distribution. Then generate matrix  $A$  randomly from the standard normal distribution, and divide it evenly into  $N$  blocks. Finally, the vector  $c$  is determined by  $c = Ax^* + \eta$ , where  $\eta \sim N(0, \delta^2 I)$  is the noise.

Now let's specify the parameters used in our algorithms. We use the same augmented Lagrangian parameter  $\gamma$  and the dual stepsize  $\rho$  for the three algorithms, and set  $\gamma = \rho = \frac{10}{\|c\|}$ . The parameters in the core function are given as follows:

- The subproblem of JVAPP for  $x_i$  is

$$x_i^{k+1} = \arg \min_{x_i} \|x_i\|_1 + \langle \lambda^k, A_i x_i \rangle + \frac{\theta_i}{2\epsilon} \|A_i x_i + \sum_{j \neq i} A_j x_j^k - c\|^2 + \frac{\alpha_i}{2\epsilon} \|x_i - x_i^k\|_{\bar{P}_i}^2 + (\gamma - \frac{\theta_i}{\epsilon}) \langle Ax^k - c, A_i x_i \rangle,$$

and we let  $P_i = I$ ,  $\theta_i = \gamma$  and  $\alpha_i = 1$ .

- The subproblem of PJVAPP for  $x_i$  is

$$x_i^{k+1} = \arg \min_{x_i} \|x_i\|_1 + \langle \lambda^k, A_i x_i \rangle + \frac{\theta_i}{2\epsilon} \|A_i x_i + \sum_{j \neq i} A_j x_j^k - c\|^2 + (\gamma - \frac{\theta_i}{\epsilon}) \langle Ax^k - c, A_i x_i \rangle,$$

and we let  $\theta_i = \gamma$ .

- The subproblem of PVAPP for  $x_i$  is

$$x_i^{k+1} = \arg \min_{x_i} \|x_i\|_1 + \langle \lambda^k, A_i x_i \rangle + \frac{\alpha_i}{\epsilon} \|x_i - x_i^k\|_{\bar{P}_i}^2 + \gamma \langle Ax^k - c, A_i x_i \rangle,$$

and we let  $P_i = I$ ,  $\alpha_i = 1$ .

Finally,  $\epsilon$  is initialized as  $\frac{\beta^k}{0.1n\rho}$  and is adaptively tuned by the strategy described in Section 4.

To make comparison, for the same problem we also report the results of PJADM proposed in [17], where the authors call it Prox-JADMM. PJADM solves the  $x_i$ -subproblems as follows:

$$\begin{aligned} x_i^{k+1} &= \arg \min_{x_i} \|x_i\|_1 + \frac{\rho}{2} \|A_i x_i + \sum_{j \neq i} A_j x_j^k - c - \frac{\lambda^k}{\rho}\|^2 + \frac{1}{2} \|x_i - x_i^k\|_{\bar{P}_i}^2 \\ &= \arg \min_{x_i} \|x_i\|_1 + \langle \rho A_i^\top (Ax^k - c - \frac{\lambda^k}{\rho}), x_i \rangle + \frac{\tau_i}{2} \|x_i - x_i^k\|^2. \end{aligned}$$

The proximal parameters  $\tau_i$  are initialized with  $0.1N\rho$  and then adaptively updated by the strategy discussed in Section 2.3 of [17].

In our experiments, we simply let all the algorithms run for a fixed number of iterations under a setting with noise and a noiseless setting. Their relative error  $\frac{\|x^k - x^*\|_2}{\|x^*\|_2}$  are plotted in Figure 2 and Figure 3 respectively. In particular, Figure 2 shows the comparison result with  $n = 1000$ ,  $m = 300$ ,  $k = 60$  and  $\delta = 0$ , while Figure 3 shows the comparison result with  $n = 1000$ ,  $m = 300$ ,  $k = 60$  and  $\delta = 10^{-3}$ . From those figures, we can see that JVAPP and PJVAPP are both faster than PVAPP. On the other hand, comparing to Jacobian type ADMM, they suggest that PJVAPP is comparable to PJADM.

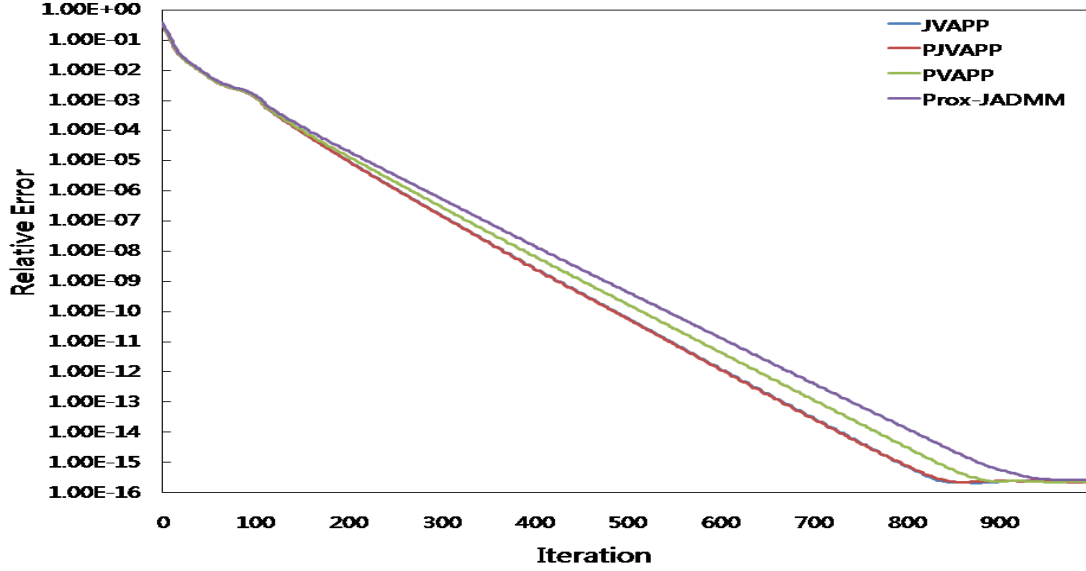


Figure 2: Convergence of VAPP by the problem (37) with noise free( $n = 1000$ ,  $m = 300$ ,  $k = 60$ ,  $\delta = 0$ )

### 6.3 Fused Logistic Regression

Note that the objective functions of the previous examples are separable. To test the efficiency of our algorithms on handling coupled objectives, we consider the following *fused logistic regression* problem:

$$\min_{x,c} \ell(x,c) + \mu \|x\|_1 + \nu \sum_{j=2}^n |x_j - x_{j-1}| \quad (38)$$

where  $x \in \mathbf{R}^n$ ,  $c \in \mathbf{R}$ , and

$$\ell(x,c) = \frac{1}{m} \sum_{i=1}^m \log(1 + \exp(-b_i(a_i^\top x + c))),$$

with  $A = [a_1, \dots, a_m]^\top \in \mathbf{R}^{m \times n}$  and  $b = [b_1, \dots, b_m]^\top \in \mathbf{R}^m$ . This problem is essentially the sparse logistic regression problem with an additional focus on the natural ordering in the features. Obviously,  $x, c$  is nonseparable in  $\ell(x, c)$ . To facilitate our analysis, we rewrite problem (38) equivalently as

$$\begin{aligned} \min_{x,c,y} \quad & \ell(x,c) + \mu \|x\|_1 + \nu \|y\|_1 \\ \text{s.t.} \quad & y = Dx. \end{aligned} \quad (39)$$

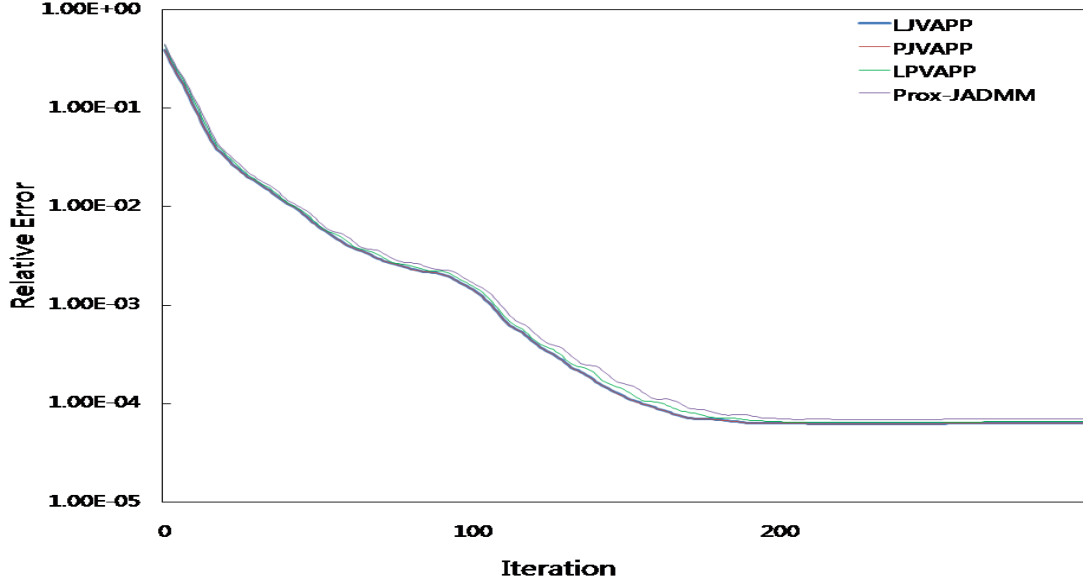


Figure 3: Convergence of VAPP by the problem (37) with noise added( $n = 1000$ ,  $m = 300$ ,  $k = 60$ ,  $\delta = 10^{-3}$ )

where  $x \in \mathbf{R}^n$ ,  $c \in \mathbf{R}$ , and  $y \in \mathbf{R}^{n-1}$ . The corresponding augmented lagrangian function is given by

$$L_\gamma(x, y, c, p) = \frac{1}{m} \sum_{i=1}^m \log(1 + \exp(-b_i(a_i^\top x + c))) + \mu \|x\|_1 + \nu \|y\|_1 + \langle p, y - Dx \rangle + \frac{\gamma}{2} \|y - Dx\|^2,$$

and the gradients of  $\ell(x, c)$  with respect to  $x$  and  $c$  are easily computed as

$$\nabla_x \ell(x, c) = -\frac{1}{m} \hat{A}^\top (1 - d), \nabla_c \ell(x, c) = -\frac{1}{m} b^\top (1 - d), d = 1. / (1 + \exp(-\hat{A}x - cb)),$$

where  $\hat{A} = [b_1 a_1, b_2 a_2, \dots, b_m a_m]^\top$  and  $1./\alpha$  denotes the component-wise division.

We apply our LPVAPP algorithm with  $P_i = I$  and  $\alpha_i = 1 \ \forall i = 1, \dots, n$ , to problem (39). At each iteration, this method runs as follow:

$$\begin{cases} x^{k+1} = \arg \min_x \langle \nabla_x \ell(x^k, c^k), x \rangle - \langle D^\top (p^k + \gamma(y^k - Dx^k)), x \rangle + \mu \|x\|_1 + \frac{1}{\epsilon} \|x - x^k\|^2 \\ c^{k+1} = \arg \min_c \langle \nabla_c \ell(x^k, c^k), c \rangle + \frac{1}{\epsilon} \|c - c^k\|^2 \\ y^{k+1} = \arg \min_y \langle p^k + \gamma(y^k - Dx^k), y \rangle + \nu \|y\|_1 + \frac{1}{\epsilon} \|y - y^k\|^2 \\ p^{k+1} = p^k + \rho(y^{k+1} - Dx^{k+1}) \end{cases}$$

We first construct a random example of size  $n = 1000$  with the regression coefficient  $\hat{x} \in \mathbf{R}^n$  chosen



as following

$$\hat{x}_j = \begin{cases} r_1, j = 101, 102, \dots, 200 \\ r_2, j = 301, 302, \dots, 400 \\ r_3, j = 501, 502, \dots, 600 \\ r_4, j = 701, 702, \dots, 800 \\ 0, \text{else} \end{cases}$$

where scalars  $r_1, r_2, r_3, r_4$  are created randomly uniform in  $(0, 20)$ . The entries of matrix  $A \in \mathbf{R}^{m \times n}$  with  $m = 500$  and  $n = 1000$  are drawn from standard normal distribution  $\mathcal{N}(0, 1)$ . Vector  $b \in \mathbf{R}^m$  is then created as the signs of  $A\hat{x} + ce$ , where  $c$  is a random number in  $(0, 1)$  and  $e$  is the  $m$ -dimensional vector of all ones. We choose  $\mu = 3 \times 10^{-2}$  and  $\nu = 5 \times 10^{-2}$  in (38). The regression result by LPVAPP is described in Figure 4, from which we can see that the natural ordering is well preserved.

To show the capability of LPVAPP on the large scaled fused logistic regression model (39), we conduct the following tests. First, we create the regression coefficient  $\hat{x} \in \mathbf{R}^n$  for  $n \geq 100$  as

$$\hat{x}_j = \begin{cases} 20, j = 1, 2, \dots, 20, \\ 30, j = 41, \\ 10, j = 71, \dots, 85, \\ 20, j = 121, \dots, 125, \\ 0, \text{else}, \end{cases}$$

which is the same test example used in [36]. The matrix  $A$  and vector  $b$  are created similarly to the previous example. Then, we apply both PVAPP and EGADM in [36] to solve the fused logistic regression problem (39) for  $m = 1000, 2000$  and  $n = 5000, 10000, 20000$ . For each setting, we generate and test 20 random instances. The average iteration number, CPU time, sparsity of  $x$  (denoted by  $\|x\|_0$ ) and sparsity of the fused term  $Lx$  (denoted by  $\|Lx\|_0$ ) of both approaches are reported in Table 1. From this table, we conclude that PVAPP is significantly faster than EGADM when the sample size of (39) is over 2000.

## 7 Conclusion

In this paper, we propose a new VAPP framework that can allow the core function of standard APP to be different at each iteration. We illustrate that how some known algorithms (especially some variants of ADMM) can be considered as specialization of our framework. We prove that under some mild conditions, the solution sequence generated by VAPP converges. In particular,

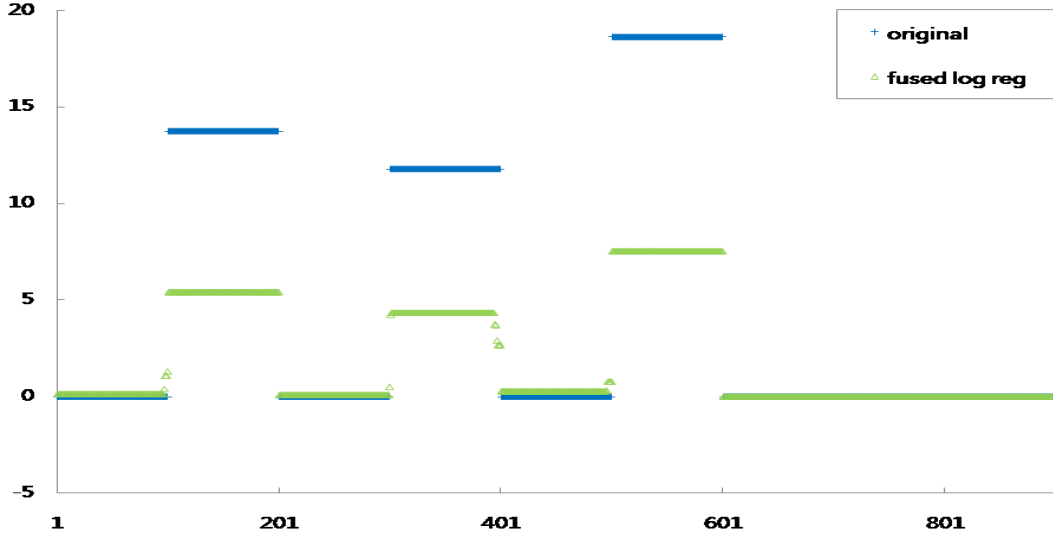


Figure 4: The regression result by the fused logistic regression model ( $n = 1000$ ,  $m = 500$ )

Table 1: Comparing results for EGADM and PVAPP

		EGADM				PVAPP			
$m$	$n$	iter	cpu	$\ x\ _0$	$\ Lx\ _0$	iter	cpu	$\ x\ _0$	$\ Lx\ _0$
1000	10000	96.65	2.3	35.75	36.1	116.7	1.4	36.55	37.65
2000	5000	94.75	2.4	30.7	26.9	115.1	1.4	30.25	30.45
2000	10000	260.95	12.5	32.4	15.75	177.25	4.2	30.02	19.9
2000	20000	120.9	11.2	34.55	34.5	125.7	5.9	34.15	34.15

the condition considered in this paper is weaker than that of JADM. When the core function is specialized to be quadratic, we show that an  $o(1/k)$  iteration complexity can be obtained. In addition, our framework can handle a convex problem with nonseparable objective and coupled linear constraints. Moreover, it can accommodate parallel computing, which in turn makes it well suited for large-scale problems. Finally, our numerical results show that the proposed VAPP has promising numerical efficiency and iteration simplicity.

## References

- [1] Auslender, A., & Teboulle, M. (2001). Entropic proximal decomposition methods for convex programs and variational inequalities. *Mathematical programming*, 91(1), 33-47.
- [2] Beck, A., & Teboulle, M. (2003). Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3), 167-175.
- [3] Beck, A., & Teboulle, M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1), 183-202.
- [4] Beck, A., & Teboulle, M. (2014). A fast dual proximal gradient algorithm for convex minimization and applications. *Operations Research Letters*, 42(1), 1-6.
- [5] Bertsekas, D.P. (1999). *Nonlinear Programming*. Athena Scientific, Belmont Massachusetts.
- [6] Bregman, L.M. (1967). The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. *USSR computational mathematics and mathematical physics*, 7(3), 200-217.
- [7] Cai, X., Han, D., & Yuan, X. (2014). The direct extension of ADMM for three-block separable convex minimization models is convergent when one function is strongly convex. *Optimization Online*.
- [8] Çela, A. S., & Hamam, Y. (1992). Optimal motion planning of a multiple-robot system based on decomposition coordination. *Robotics and Automation, IEEE Transactions on*, 8(5), 585-596.
- [9] Chen, C., He, B., Ye, Y., & Yuan, X. (2014). The direct extension of ADMM for multi-block convex minimization problems is not necessarily convergent. *Mathematical Programming*, 1-23.
- [10] Chen, G., & Teboulle, M. (1993). Convergence analysis of a proximal-like minimization algorithm using Bregman functions. *SIAM Journal on Optimization*, 3(3), 538-543.
- [11] Censor, Y., & Zenios, S. A. (1992). Proximal minimization algorithm with D-functions. *Journal of Optimization Theory and Applications*, 73(3), 451-464.
- [12] Cohen, G. (1978). Optimization by decomposition and coordination: a unified approach. *Automatic Control, IEEE Transactions on*, 23(2), 222-232.
- [13] Cohen, G. (1980). Auxiliary problem principle and decomposition of optimization problems. *Journal of optimization Theory and Applications*, 32(3), 277-305.

- [14] Cohen, G., & Zhu, D. L. (1984). Decomposition coordination methods in large scale optimization problems. The nondifferentiable case and the use of augmented Lagrangians. *Advances in large scale systems*, 1, 203-266.
- [15] Contreras, J., Losi, A., Russo, M., & Wu, F. F. (2000). DistOpt: A software framework for modeling and evaluating optimization problem solutions in distributed environments. *Journal of Parallel and Distributed Computing*, 60(6), 741-763.
- [16] Cui, Y., Li, X., Sun, D., & Toh, K. C. (2015). On the convergence properties of a majorized ADMM for linearly constrained convex optimization problems with coupled objective functions. *arXiv preprint arXiv:1502.00098*.
- [17] Deng, W., Lai, M. J., Peng, Z., & Yin, W. (2013). Parallel multi-block ADMM with  $O(1/k)$  convergence. *arXiv preprint arXiv:1312.3040*.
- [18] Eckstein, J., & Bertsekas, D. P. (1992). On the DouglasRachford splitting method and the proximal point algorithm for maximal monotone operators. *Mathematical Programming*, 55(1-3), 293-318.
- [19] Eckstein, J. (1993). Nonlinear proximal point algorithms using Bregman functions, with applications to convex programming. *Mathematics of Operations Research*, 18(1), 202-226.
- [20] Fortin, M., & Glowinski, R. (1983). Chapter III on decomposition-coordination methods using an augmented lagrangian. *Studies in Mathematics and Its Applications*, 15, 97-146.
- [21] Gabay, D., & Mercier, B. (1976). A dual algorithm for the solution of nonlinear variational problems via finite element approximation. *Computers & Mathematics with Applications*, 2(1), 17-40.
- [22] Gao, X., & Zhang, S. (2015). First-Order Algorithms for Convex Optimization with Nonseparate Objective and Coupled Constraints. *Working Paper*.
- [23] Glowinski, R., & Marroco, A. (1975). Sur l'approximation par éléments finis et la résolution par pénalisation-dualité d'une classe de problèmes de dirichlet non linéaires, *RAIRO*, 9(2), 41C76.
- [24] Glowinski, R. (1984). Decomposition-Coordination Methods by Augmented Lagrangian: Applications. In *Numerical Methods for Nonlinear Variational Problems* (pp. 166-194). Springer Berlin Heidelberg.
- [25] Glowinski, R., & Le Tallec, P. (1989). *Augmented Lagrangian and operator-splitting methods in nonlinear mechanics* (Vol. 9). SIAM.

- [26] Glowinski, R. (2014). On alternating direction methods of multipliers: a historical perspective. In *Modeling, Simulation and Optimization for Science and Technology* (pp. 59-82). Springer Netherlands.
- [27] Han, D., & Yuan, X. (2012). A note on the alternating direction method of multipliers. *Journal of Optimization Theory and Applications*, 155(1), 227-238.
- [28] Harker, P. T., & Pang, J. S. (1990). Finite-dimensional variational inequality and nonlinear complementarity problems: a survey of theory, algorithms and applications. *Mathematical programming*, 48(1-3), 161-220.
- [29] He, B., & Yuan, X. (2012). On the  $O(1/n)$  Convergence Rate of the Douglas-Rachford Alternating Direction Method. *SIAM Journal on Numerical Analysis*, 50(2), 700-709.
- [30] He, B., Hou, L., & Yuan, X. (2013). On full Jacobian decomposition of the augmented Lagrangian method for separable convex programming. *Preprint*.
- [31] Hong, M., Chang, T. H., Wang, X., Razaviyayn, M., Ma, S., & Luo, Z. Q. (2014). A block successive upper bound minimization method of multipliers for linearly constrained convex optimization. *arXiv preprint arXiv:1401.7079*.
- [32] Hong, M., & Luo, Z. Q. (2012). On the linear convergence of the alternating direction method of multipliers. *arXiv preprint arXiv:1208.3922*.
- [33] Kiwiel, K. C. (1997). Proximal minimization methods with generalized Bregman functions. *SIAM journal on control and optimization*, 35(4), 1142-1168.
- [34] Kim, B. H., & Baldick, R. (1997). Coarse-grained distributed optimal power flow. *IEEE Transactions on Power Systems*, 12(2), 932-939.
- [35] Kim, B. H., & Baldick, R. (2000). A comparison of distributed optimal power flow algorithms. *Power Systems, IEEE Transactions on*, 15(2), 599-604.
- [36] Lin, T. Y., Ma, S. Q., & Zhang, S. Z. (2015). On the sublinear convergence rate of multi-block ADMM. *Journal of the Operations Research Society of China*, 3(3), 251-274.
- [37] Losi, A., & Russo, M. (2003). On the application of the auxiliary problem principle. *Journal of optimization theory and applications*, 117(2), 377-396.
- [38] Ortega, J. M., & Rheinboldt, W. C. (1970). *Iterative solution of nonlinear equations in several variables* (Vol. 30). Siam.
- [39] Renaud, A. (1993). Daily generation management at Electricit de France: from planning towards real time. *Automatic Control, IEEE Transactions on*, 38(7), 1080-1093.

- [40] Rockafellar, R. T. (2015). *Convex analysis*. Princeton university press.
- [41] Rockafellar, R. T. (1976). Augmented Lagrangians and applications of the proximal point algorithm in convex programming. *Mathematics of operations research*, 1(2), 97-116.
- [42] Shefi, R., & Teboulle, M. (2014). Rate of convergence analysis of decomposition methods based on the proximal method of multipliers for convex minimization. *SIAM Journal on Optimization*, 24(1), 269-297.
- [43] Teboulle, M. (1992). Entropic proximal mappings with applications to nonlinear programming. *Mathematics of Operations Research*, 17(3), 670-690.
- [44] Teboulle, M. (1997). Convergence of proximal-like algorithms. *SIAM Journal on Optimization*, 7(4), 1069-1083.
- [45] Zhu, D.L. (1983). *OPTIMISATION SOUS-DIFFERENTIABLE ET METHODES DE DECOMPOSITION* (Doctoral dissertation).
- [46] Zhu, D., & Marcotte, P. (1995). Coupling the auxiliary problem principle with descent methods of pseudoconvex programming. *European journal of operational research*, 83(3), 670-685.
- [47] Zhu, D. L. (2003). Augmented Lagrangian theory, duality and decomposition methods for variational inequality problems. *Journal of optimization theory and applications*, 117(1), 195-216.